Sprint Review and Retrospective

22EW5 CS-250

Jared Smith

Southern New Hampshire University

In the course of developing the SNHU Travel project, our team utilized a development methodology known as Scrum-agile. We organized ourselves according to the team structure defined by scrum and attempted to adhere to the principles and practices of Agile to achieve the most efficient and client-focused development solutions for the project. As the Scrum Master, it was my job to first and foremost organize the team. We assigned a product owner, a developer, and a tester. Each of these roles are crucial to the Scrum team, and their individual contributions and roles will be expounded further in this report.

Next, in my role as the Scrum Master, it was my responsibility to organize and lead any Scrum event, including our planning meeting, daily Scrum meetings, and any other meetings the team deems necessary to complete the project. Most importantly, I want to highlight the daily Scrum meetings, which I value quite highly. The daily Scrum is where each team member reports what they’ve done since the previous day’s meeting, outlines anything they need to continue their work, and lets the team know what they’ll be working on over the course of the next day. These meetings are crucial for team organization, because Scrum-agile encourages a collaborative approach, with each member of the team involved in each process of the development cycle.

The Product Owner is another essential member of the Scrum team, who’s primary responsibility is acting as a liaison between the Scrum team and the client. The Product Owner meets with the client to get a solid definition of their needs and expectations for the project. It was then their job to ensure that the client’s needs are understood by the scrum team, and the client understands the limits of what the team could accomplish in the development window. From there, the Product Owner managed our product backlog, helping us prioritize essential functionalities and client requirements. It was also their job to be available to the Scrum Team to receive clarification from the client, and to be available to the client in case of changing needs. In the course of the SNHU Travel project, our Product Owner received word from the client that they wanted some of the functionality of their site to be adjusted to show a specific kind of vacations, which they had just learned would be their core upcoming market. It was the Product Owner who notified the team, and helped organize the changes that were necessary to adjust the project to meet client needs.

The Developers are the backbone of the Scrum Team, the team members who actually write the code and develop the project as requested. Our developer, in addition to being able to develop the client’s software, needed to have excellent communication skills during this project. Some of the client requests began to seem a bit vague when it came to implementation, so our developer had to request information and clarification from our Product Owner, while still focusing on completing the work he could with the information he had. He also had to communicate regularly with our tester, in order to get clear and specific feedback about the functionality of the code.

Lastly, our Tester was responsible for ensuring code was functional and met client standards. This was accomplished by defining specific test cases for each user story in the product backlog, that has specific approval criteria. Then, when products were submitted by the developers, the tester measured the output against the test cases to determine whether the code was acceptable or not. In the SNHU travel project, our tester had an issue with unclear test cases, which was remedied by collaboration with the Product Owner. This produced high quality test cases that made ensuring client satisfaction much easier.

One of the big hurdles in this project was a mid-sprint change in client requirements. We received word that the Client wanted us to focus our efforts on a certain kind of vacation, which was a more specific requirement than the one we were working on at the time. This kind of change requires effective communication throughout the team, which was accomplished mostly by Email. Our tester emailed our product owner to get more specific requirements for the team, our developer emailed our tester to ensure that the new test cases were clear, and our product owner collaborated with the rest of the team to ensure that the new client requirements were understood by all. Overall, the Scrum-agile approach encouraged our team to be collaborative in meeting the new client needs.

As outlined in the beginning of this report, we closely followed the Scrum-agile organization and principles. According to the Scrum guidelines, we kept our team small to facilitate communication and collaboration. We also held daily Scrum meetings to keep the team well informed and we utilized affinity grouping, an Agile estimation method, to organize our product backlog. These methodologies were highly effective for our team, and helped to keep us on track, even with unexpected setbacks and changes.

Scrum-agile was a very good choice for our team, and gave us the tools we needed to accomplish the task, and deal with any obstacles that arose during the development cycle. The pros of this approach for our team include a solid organization, easy communication, a good structure for estimating our project workload and well-defined roles within the team. The only con I can see with this approach for the SNHU Travel project was a sometimes-overzealous planning and organization. At times during the project, it felt as though we spent substantially more man hours planning and organizing than developing a project. Overall, though, I would say Scrum-agile was an effective approach for the SNHU Travel project.

As far as whether Scrum-agile was the best approach for the SNHU Travel project, I can’t see how any other development method in current practice among the dev community could have improved our work on this project. Waterfall would have been a disaster the moment the client changed requirements on us. If we had developed a large section of non-functioning code that focused entirely on previous requirements, we would not have been able to change course so quickly. This would have resulted in either having to tell the client we couldn’t fill their requirements, or that we would need more time for the project. So, I think I can definitively say that Scrum-agile was the best approach.